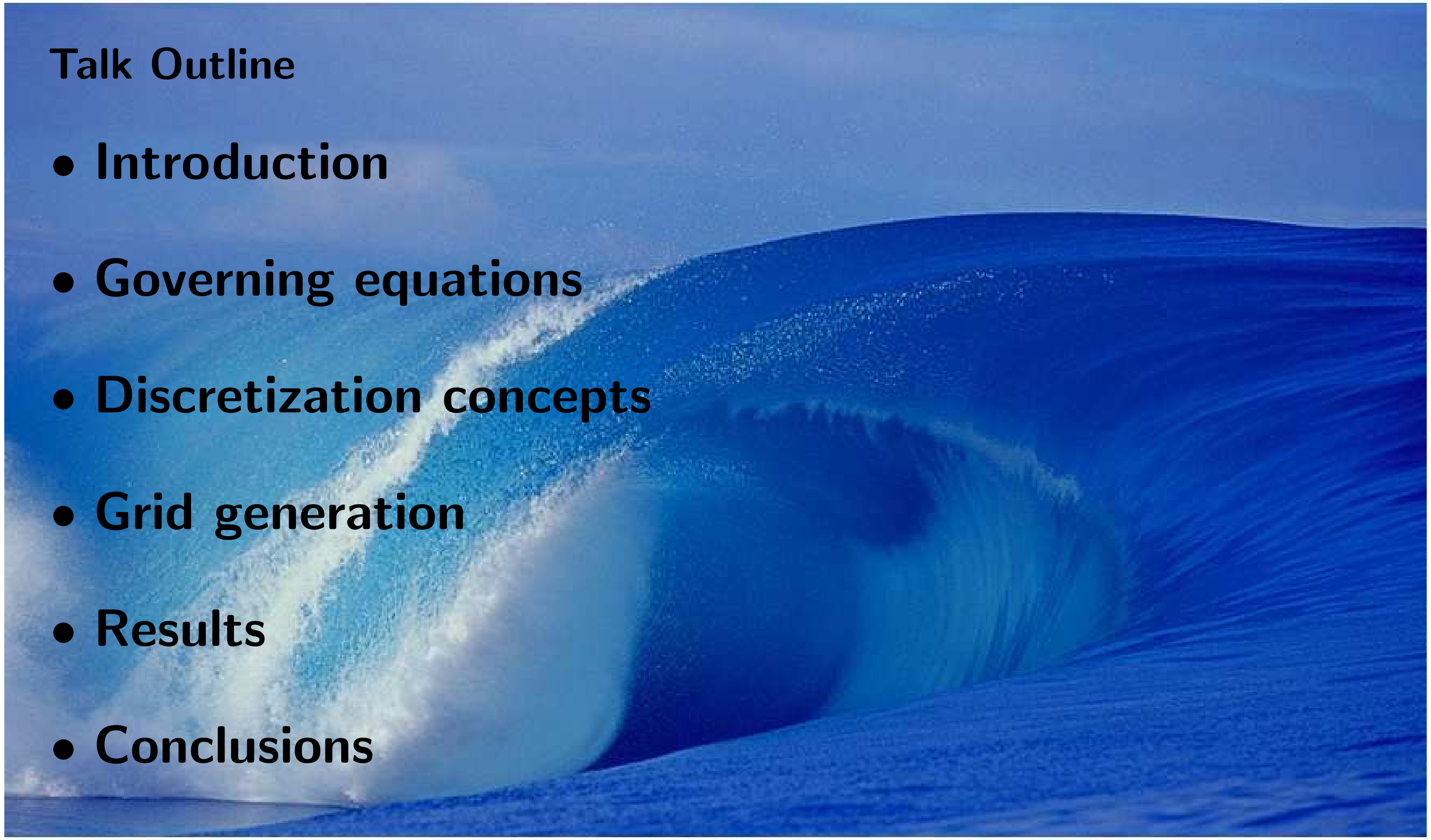
An Embedded Boundary Adaptive Mesh Refinement Method for Environmental Flows

Mike Barad

Civil and Environmental Engineering
University of California, Davis

EFMH Seminar
Stanford University
May 16, 2005



Talk Outline

- **Introduction**
- **Governing equations**
- **Discretization concepts**
- **Grid generation**
- **Results**
- **Conclusions**

Introduction

What types of environmental flows do we hope to analyze with this method?

- Highly non-linear, multi-scale flows in oceans, lakes, and rivers
- Flows that are well approximated by the variable density incompressible Navier-Stokes equations
- Examples: internal waves, coastal plumes, density currents in lakes, flows in branched estuarine slough networks, flows past highly complex topography

What are the issues involved?

- Complex and often sparse geometries
- Large ranges in spatial and temporal scales
- Moving fronts and highly complex mixing zones

What do we hope to provide with such a tool?

- An enhanced ability to interpret and extend the results of field and laboratory studies
- A predictive tool for both engineering and science

Variable Density Incompressible Navier-Stokes Equations

- Momentum balance

$$\vec{u}_t + (\vec{u} \cdot \nabla)\vec{u} = -\frac{\nabla p}{\rho} + \vec{g} + \nu \Delta \vec{u}$$

- Divergence free constraint

$$\nabla \cdot \vec{u} = 0$$

- Density conservation

$$\rho_t + \vec{u} \cdot \nabla \rho = 0$$

- Passive scalar transport

$$c_t + \vec{u} \cdot \nabla c = \nabla \cdot (k_c \nabla c) + H_c$$

Note that we do not employ Boussinesq or hydrostatic approximations.

Solution Strategy: Temporal Discretization

We build on a classic second-order accurate projection method (Bell, Colella, Glaz, JCP 1989). We split the momentum equations into three pieces:

- Hyperbolic: $\vec{u}_t + (\vec{u} \cdot \nabla)\vec{u} = H$

  where we exactly enforce a divergence free state for the advective velocities, and compute the advective term explicitly

- Parabolic: $\vec{u}_t = \nu \Delta \vec{u} + S$

  which we solve implicitly for a predictor velocity

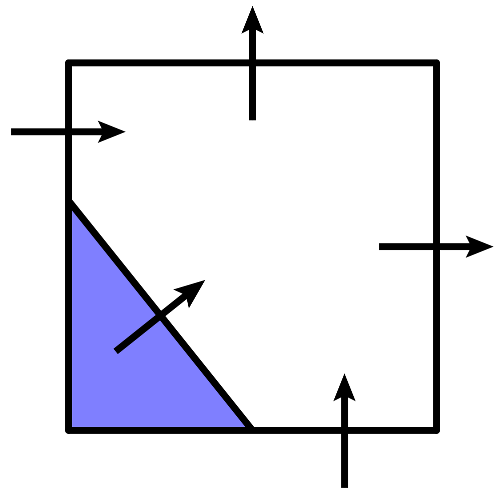
- Elliptic: $\nabla \cdot \frac{1}{\rho} \nabla p = \nabla \cdot (-(\vec{u} \cdot \nabla)\vec{u} + \nu \Delta \vec{u})$

  which we solve implicitly for pressure, and subsequently correct the predictor velocity

To update the scalar equations we do similar hyperbolic and parabolic decompositions.

Solution Strategy: Spatial Discretization Using Embedded Boundaries (EB)

For the bulk of the flow, $O(n^3)$ cells in 3D, we compute on a regular Cartesian grid. We use an embedded boundary description for the $O(n^2)$ (in 3D) control-volumes that intersect the boundary.



Advantages of underlying rectangular grid:

• Grid generation is tractable, with a straightforward coupling to block-structured adaptive mesh refinement (AMR)

• Good discretization technology, e.g. well-understood consistency theory for finite differences, geometric multigrid for solvers.

EB Finite-Volume Discretization Concepts

- Consider hyperbolic and elliptic conservation laws:

$$\frac{\partial U}{\partial t} + \nabla \cdot \vec{F}(U) = S \qquad \text{and} \qquad \nabla \cdot \nabla \phi = \nabla \cdot \vec{F} = H$$

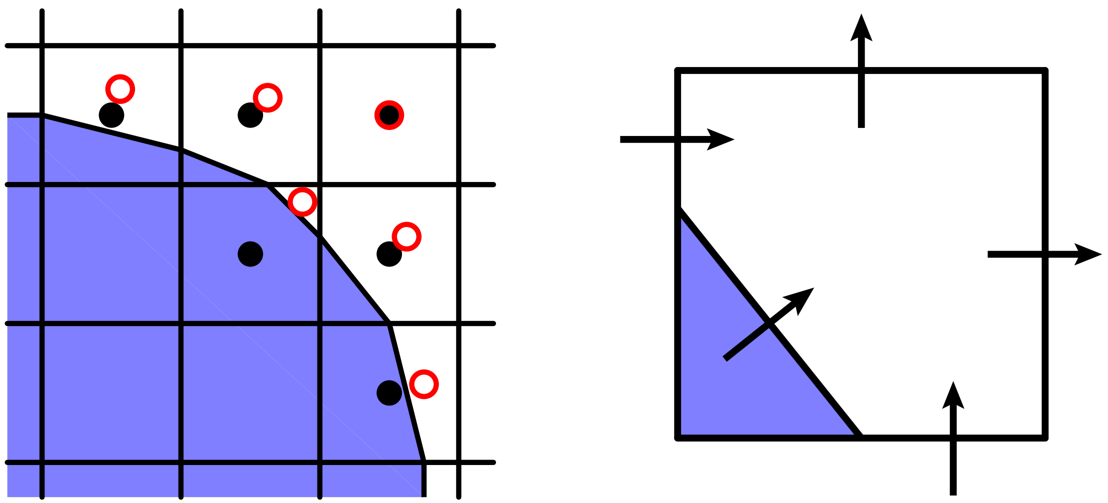
- Primary dependent variables approximate values at centers of Cartesian cells

- Divergence theorem over each control volume leads to a "finite volume" approximation for $\nabla \cdot \vec{F}$ (fluxes are at centroids):

$$\nabla \cdot \vec{F} \approx \frac{1}{\kappa \Delta x^D} \int \nabla \cdot \vec{F} dx = \frac{1}{\kappa \Delta x} \sum \alpha_s \vec{F}_s \cdot \vec{n}_s + \alpha_B \vec{F} \cdot \vec{n}_B \equiv D \cdot \vec{F}^c$$

  Away from boundaries, our method reduces to a standard conservative finite difference method

- Given κ, α_s, α_B, and $\vec{n}_B$ on a fine grid, we can compute these on coarser grids without reference to the original geometry.



EB Finite-Volume Discretization Concepts, cont.

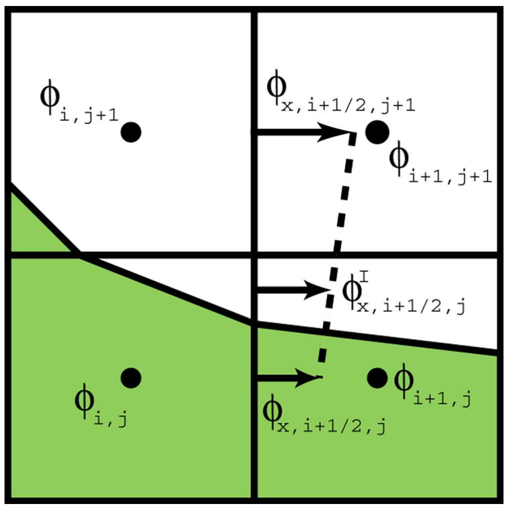
For the case of an elliptic conservation law, we need to compute $\nabla\phi$ at face centroids. Here is how we do this in 2D:

- First we compute face-centered gradients using centered differences

  $\phi_{x,i+1/2,j} = \frac{\phi_{i+1,j}-\phi_{i,j}}{\Delta x}$

  $\phi_{x,i+1/2,j+1} = \frac{\phi_{i+1,j+1}-\phi_{i,j+1}}{\Delta x}$

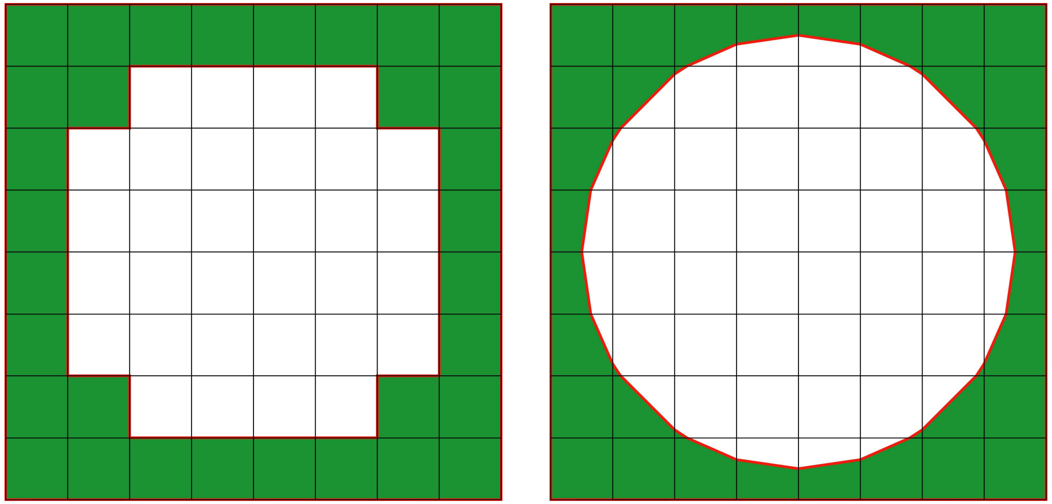
- Then we linearly interpolate the gradient (or flux) to the face centroid, yielding $\phi^I_{x,i+1/2,j}$



- In 3D we use a bilinear interpolation based on four face-centered gradients (fluxes)

Why are Embedded Boundaries Important?

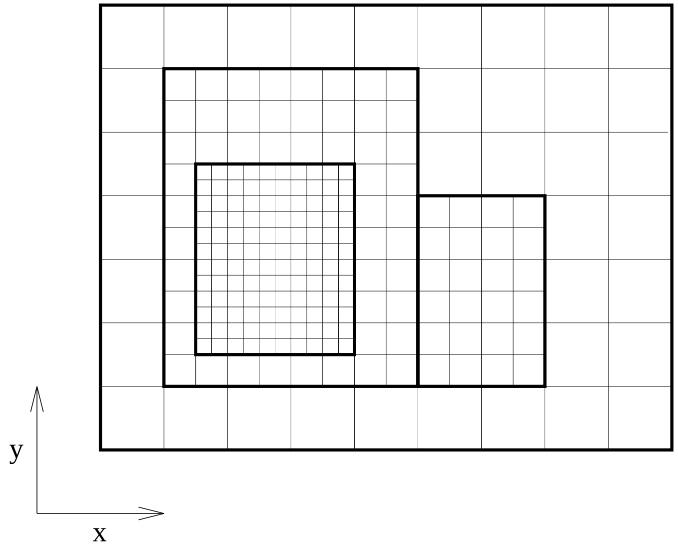
To accurately discretize our conservation laws all we need are the following quantities: volume fractions, area fractions, centroids, boundary areas, and boundary normals.



Above are two approximations to a circle: stair-step on left, and EB on right.

	Stair-Step	Embedded Boundary
Area Error	$O(h)$	$O(h^2)$
Perimeter Error	$O(1)$	$O(h^2)$
Boundary Normal Errors	$O(1)$	$O(h^2)$

Block-Structured Adaptive Mesh Refinement



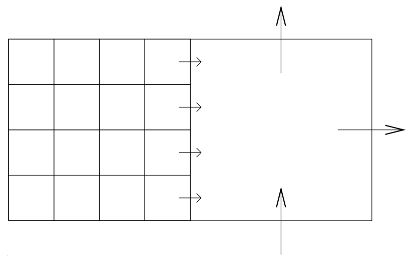
In adaptive methods, one adjusts the computational effort locally to maintain a uniform level of accuracy throughout the problem domain.

- Refined regions are organized into rectangular patches. Refinement is possible in both space and time.
- AMR allows the simulation of a range of spatial and temporal scales. Capturing these ranges is critical to accurately modeling multi-scale transport complexities such as boundaries, fronts, and mixing zones that exist in natural environments.
- We maintain accuracy and strict conservation with embedded boundaries and AMR

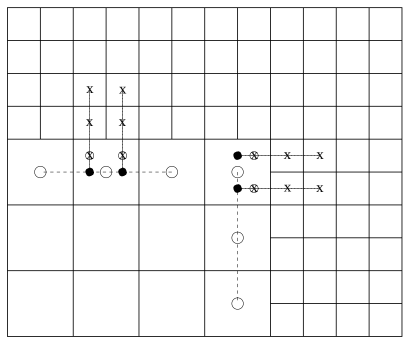
Finite-Volume Discretization Concepts

- Two requirements are necessary to maintain conservation and second-order accuracy with AMR:

  1) Match fluxes conservatively at coarse fine interfaces (this leads to a refluxing step for the coarse levels)

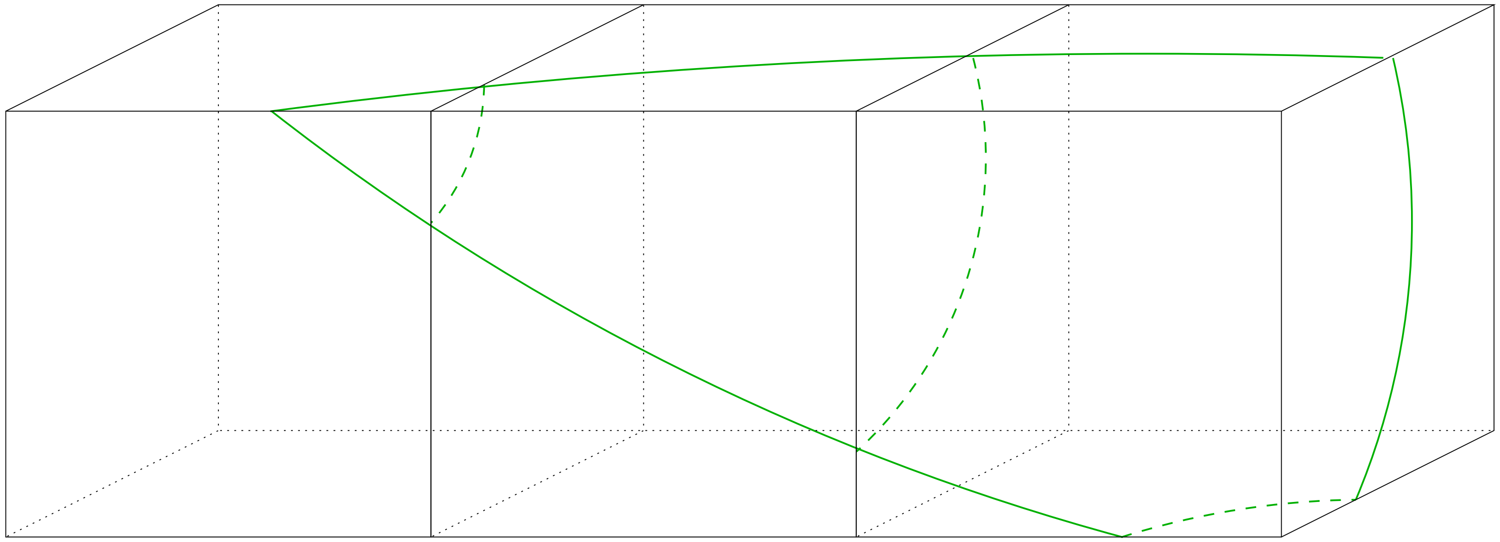


  2) Use quadratic interpolation to provide ghost cell values for points in the stencil extending outside of the grids at that level



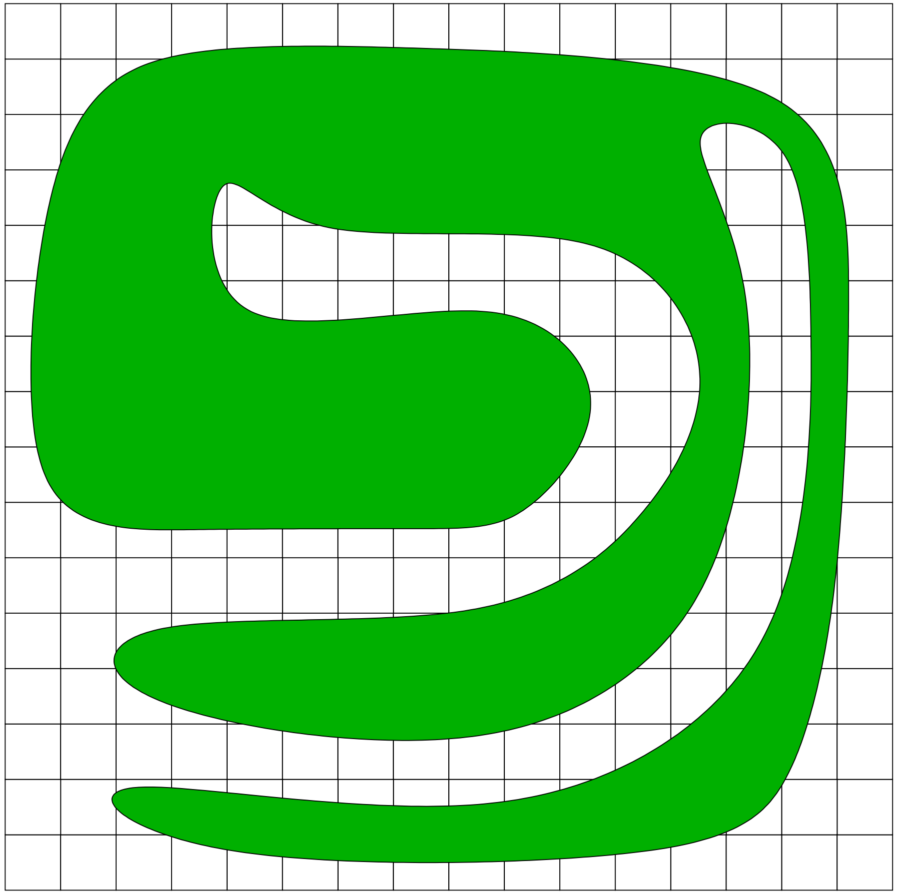
Grid Generation for Embedded Boundaries

- Three example irregular cells are shown below. Green curves indicate the intersection of the exact boundary with a Cartesian cell. We approximate face intersections using quadratic interpolants.



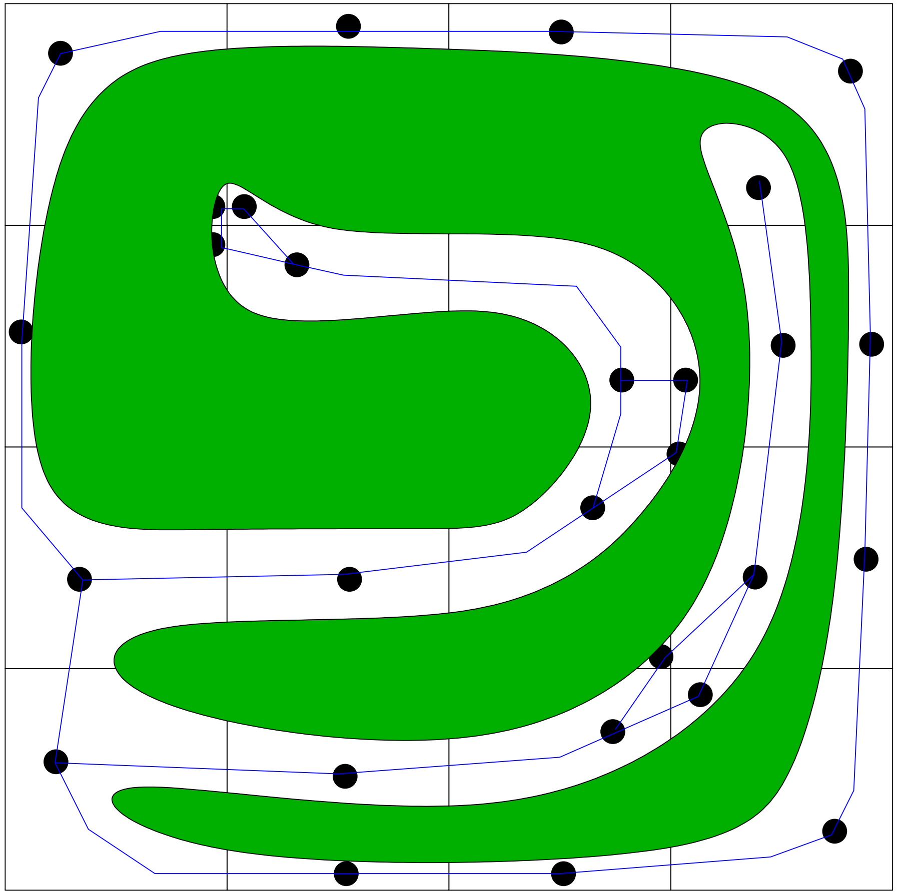
- EB's are "water tight" by construction, i.e. if two control-volumes share a face, they both have the same area fraction for that face.
- Unlike typical discretization methods, the EB control volumes naturally fit within easily parallelized disjoint block data structures.
- Permits dynamic coarsening and refinement of highly complex geometry as a simulation progresses.

EB Coarsening



White regions represent individual control volumes. Green is the exact geometry.

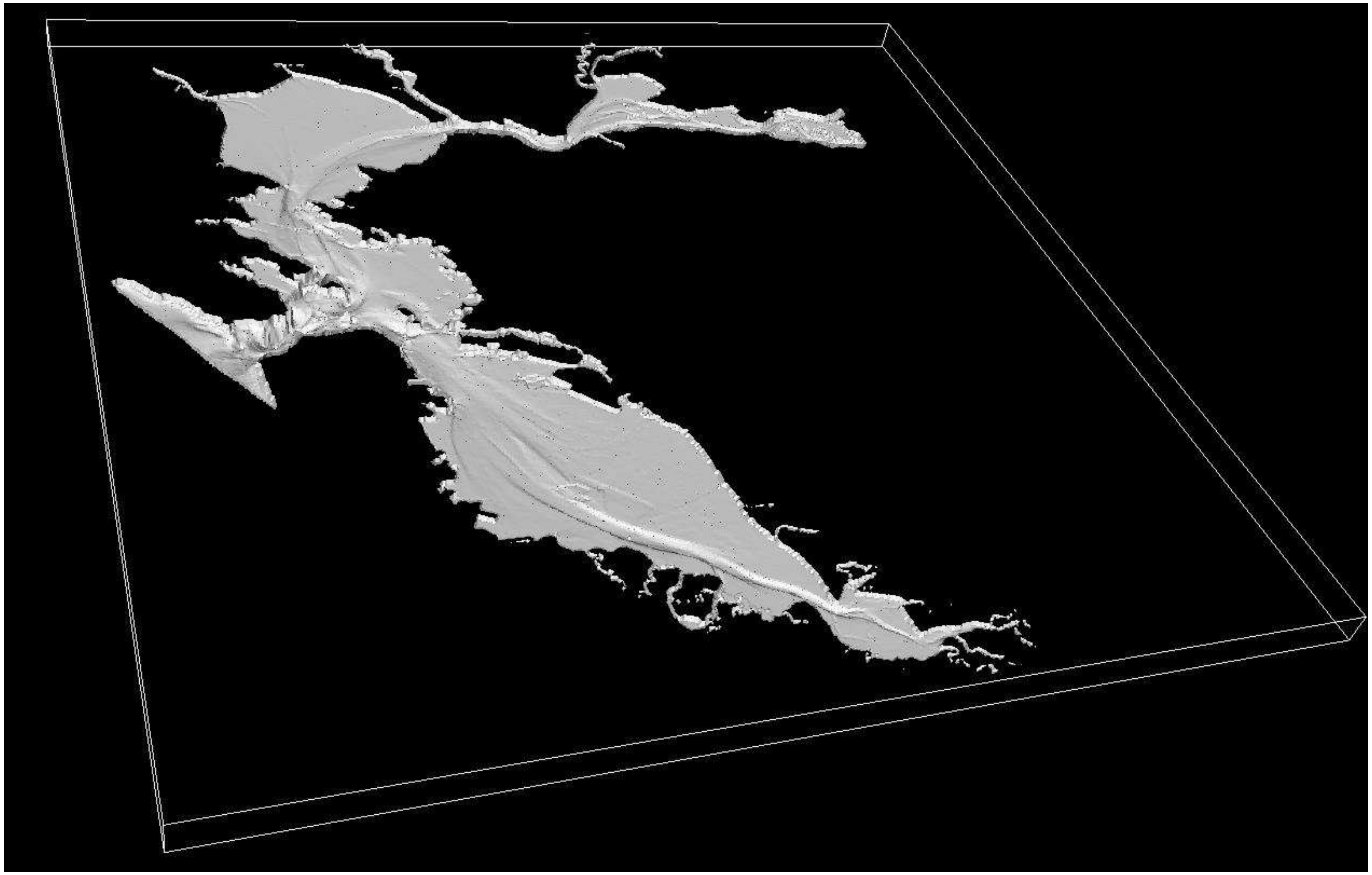
EB Coarsening



Note: We have some cells with more than one control volume per cell.

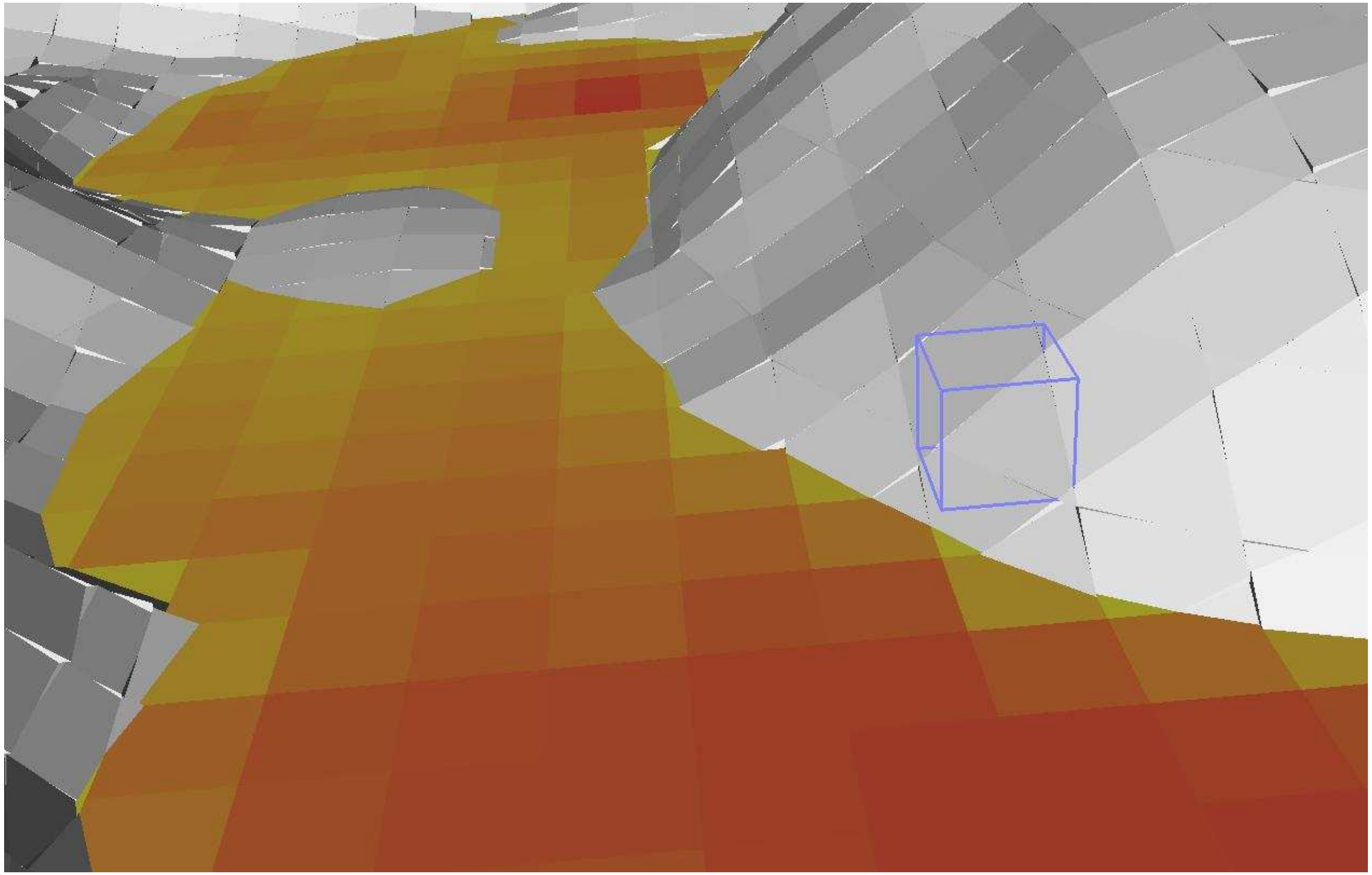
EB Grid Generation Examples

- Embedded boundary description of San Francisco Bay (using USGS DEM data)



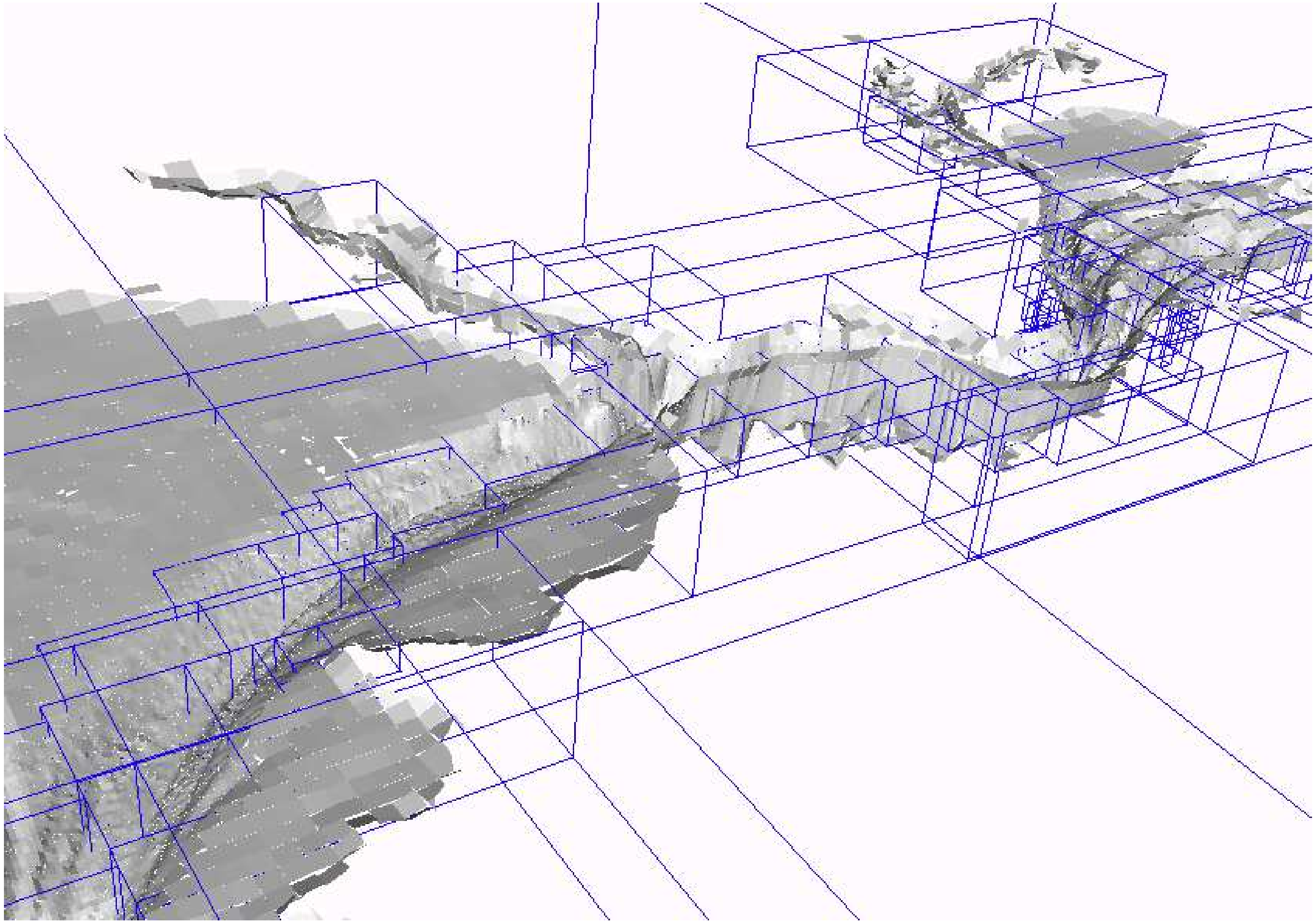
EB Grid Generation Examples

- Embedded boundary description of San Francisco Bay (using USGS DEM data)



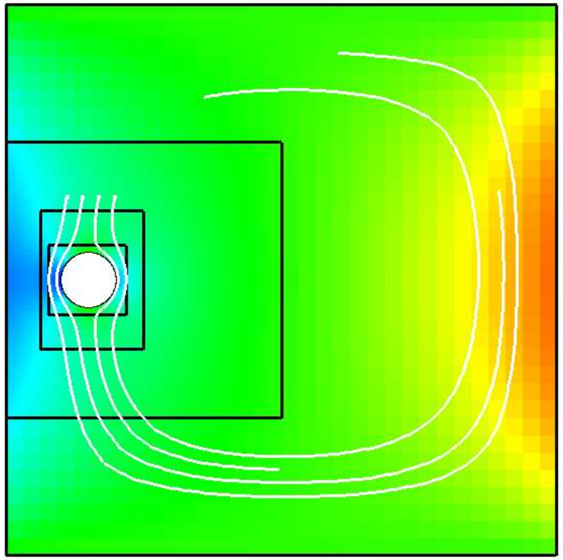
EB AMR Grid Generation Examples

- San Francisco Bay with AMR (using USGS DEM data)



Results: 2D Convergence Study

- Flow is inside a 1 m square tank, with a cylinder of diameter 0.1m at (0.15,0.5)
- The initial velocity field is the divergence free part of a rigid rotation (flow is counter-clockwise)



Above is a plot of the initial conditions. The colors indicate vertical velocity (red is up, blue is down)

- The $Re = 100$ for this problem.
- We initialize our density field as $\rho = 1000 + 30y$ (linear, with heavy on top)

Results: 2D Convergence Study

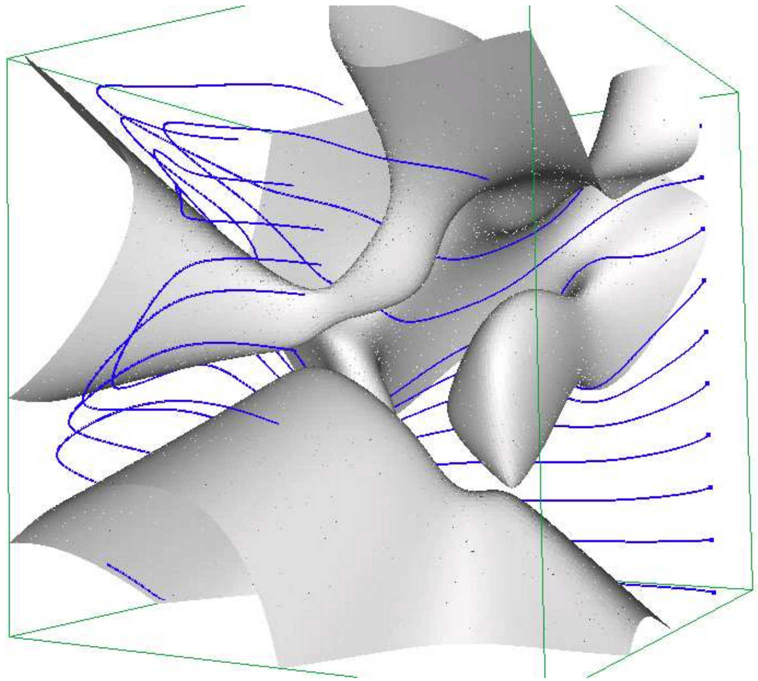
- For the convergence study, we solve this problem on 3 different grid hierarchies. We discretize the coarsest level of each hierarchy with 8x8, 16x16, and 32x32 cells.
- For each run we use 4 levels of AMR with a refinement ratio of 4 between levels
- In the table we present norms of the solution error, and our convergence rate.

Base Grids	8-16	Rate	16-32
L_1 Norm of V Velocity Error	5.73e-3	**2.02**	1.42e-3
L_2 Norm of V Velocity Error	7.68e-3	**2.01**	1.91e-3
L_1 Norm of Scalar Error	1.24e-2	**2.04**	3.03e-3
L_2 Norm of Scalar Error	1.64e-2	**1.98**	4.16e-3

Results: 3D Convergence Study

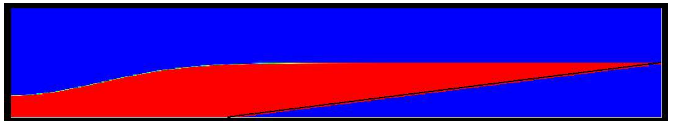
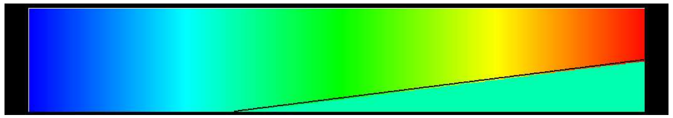
- Below is a 3D single level convergence study for a constant density, $Re = 100$, rotational flow past a complex geometry:

Base Grids	16-32	Rate	32-64
L_1 Norm of U Velocity Error	1.69e-2	**2.32**	3.39e-3
L_2 Norm of U Velocity Error	5.28e-2	**1.76**	1.55e-2
L_1 Norm of W Velocity Error	1.48e-2	**2.29**	3.03e-3
L_2 Norm of W Velocity Error	4.69e-2	**1.83**	1.32e-2



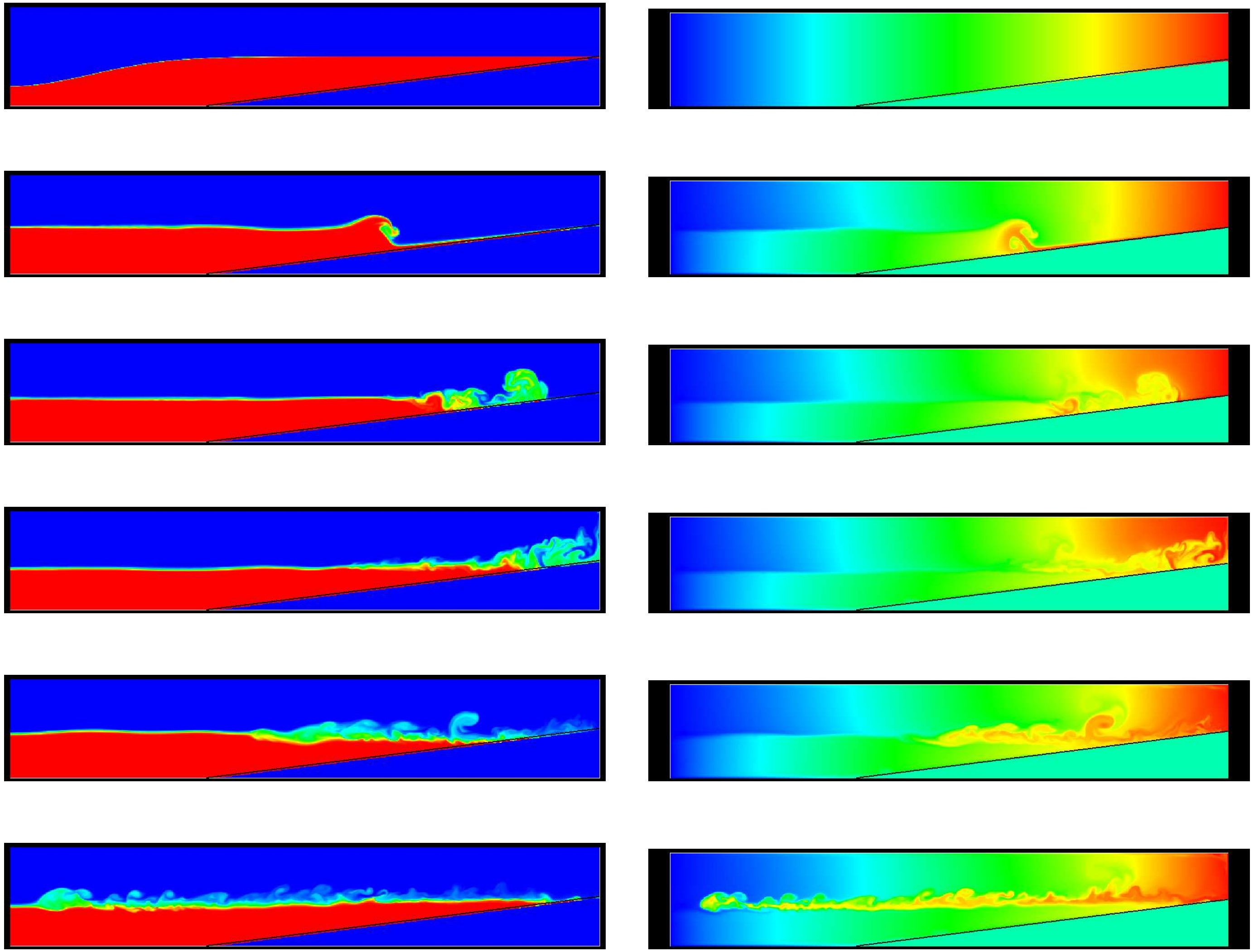
Results: Breaking Internal Waves on a Slope

- Flow is inside a 0.5m tall, by 3m wide tank, with an 8:1 slope starting 1m from the left side
- Below left is the initial density distribution (blue is light fluid, red is heavy fluid), below right is the initial conditions for a passive scalar

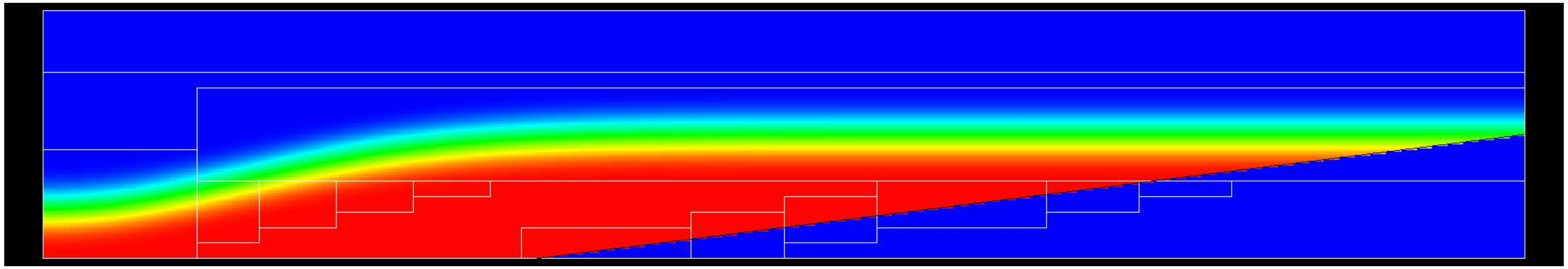
- Density ratio of light fluid to heavy fluid is 1000/1030, and our pycnocline is a step-function. The pycnocline is perturbed on the left side of the tank.
- Thanks to Prof. Fringer for this test problem

Breaking Internal Wave on a Slope (Density left, Scalar right)



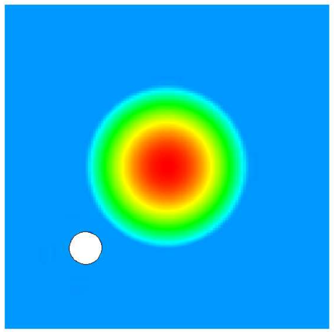
Results: Breaking Internal Waves on a Slope

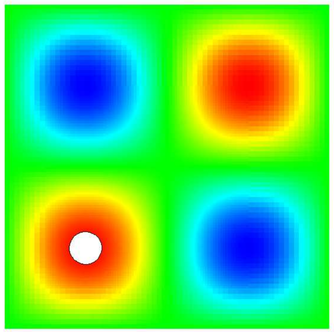
- Since most pycnoclines are not step-functions, what happens if our initial pycnocline is smoothed over 20 percent of the depth?



Results: Simulation of flow past a cylinder with AMR

- Flow is inside a 1m square tank, with a cylinder of diameter 0.1m placed at (0.25,0.25)
- We initialize the flow with a vortex patch in the center of the domain, see below left. The $Re = 100$ for this flow. We initialize a passive scalar, see below right.

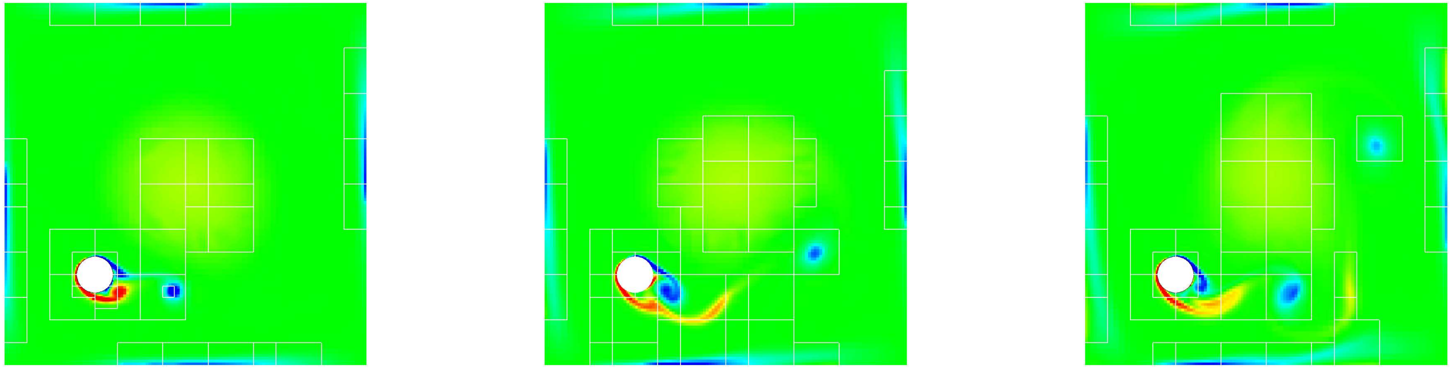




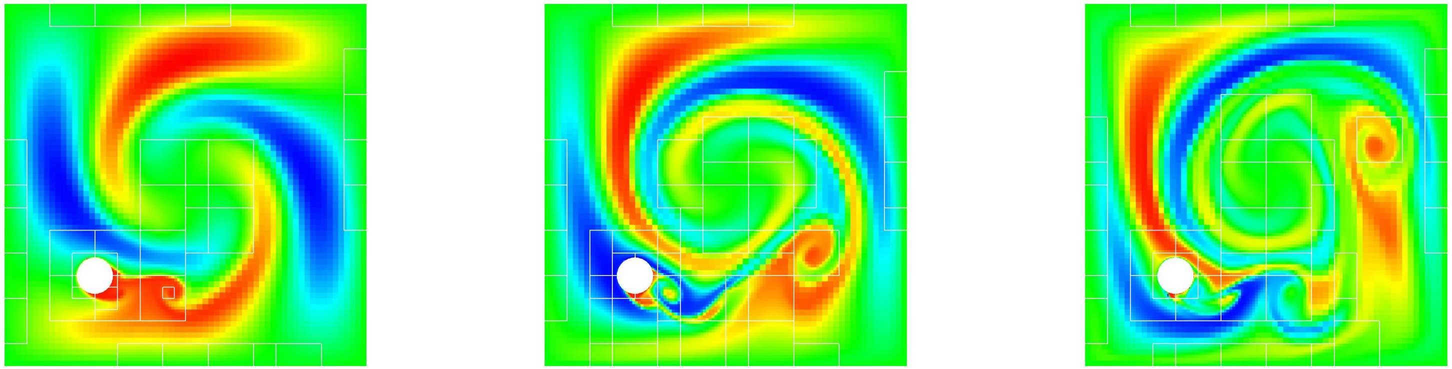
- We discretize the domain with a base grid of 64x64 cells, and refine 2 additional levels to track vorticity

Simulation of flow past a cylinder with AMR

- Vorticity (which we track with AMR):

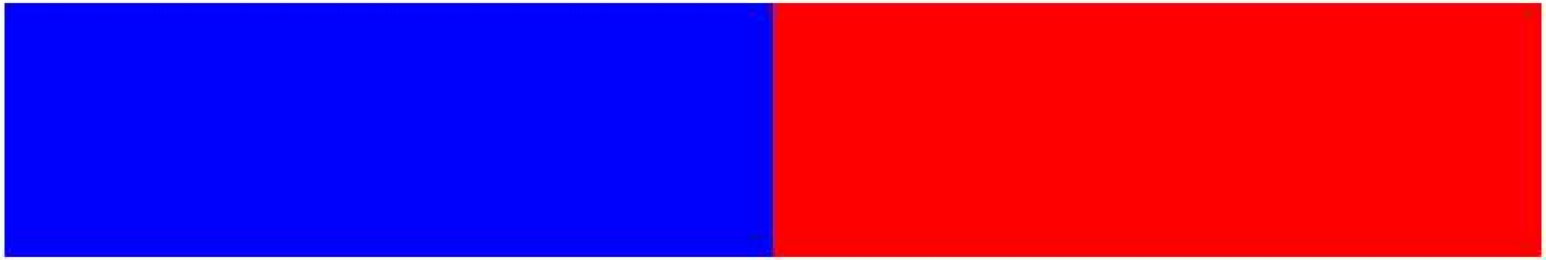


- Passive Scalar:



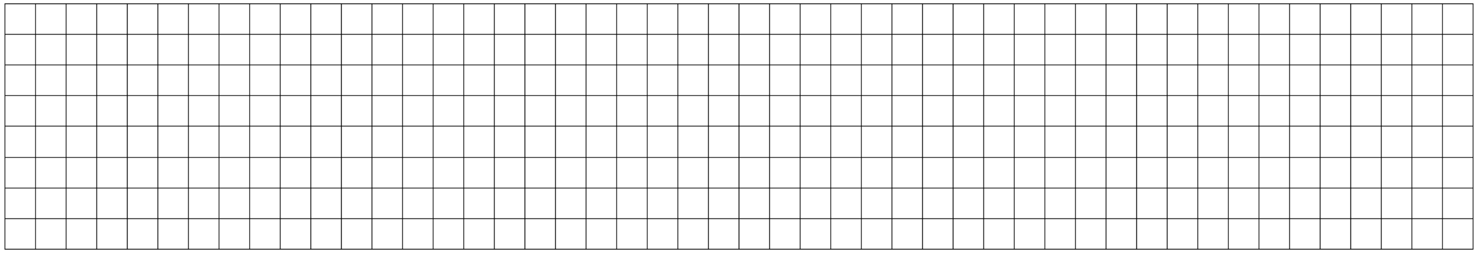
Results: Lock-Exchange with AMR

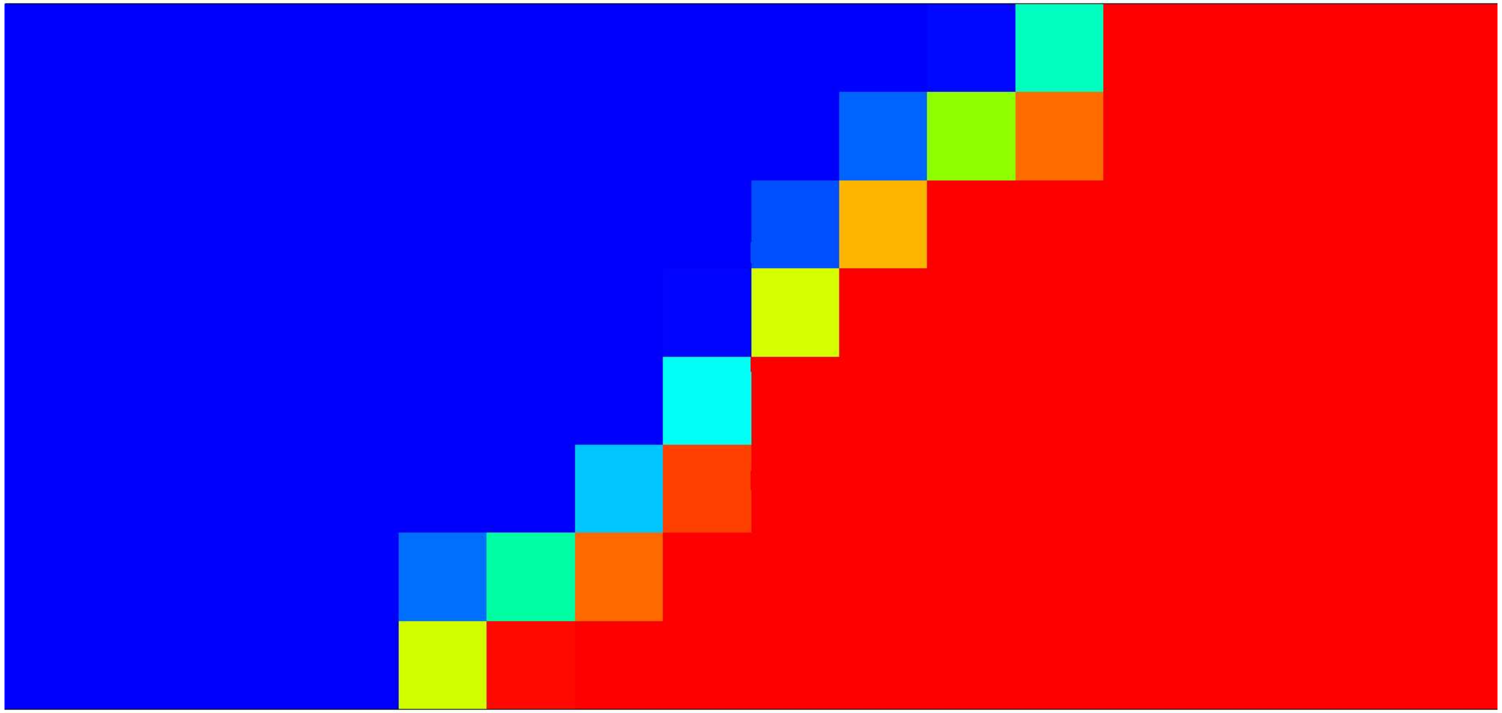
- Flow is inside a 0.5m tall, by 3m wide tank.
- On the left side of the tank we start with light water, on the right is heavy water. The density ratio of light fluid to heavy fluid is 1000/1030.



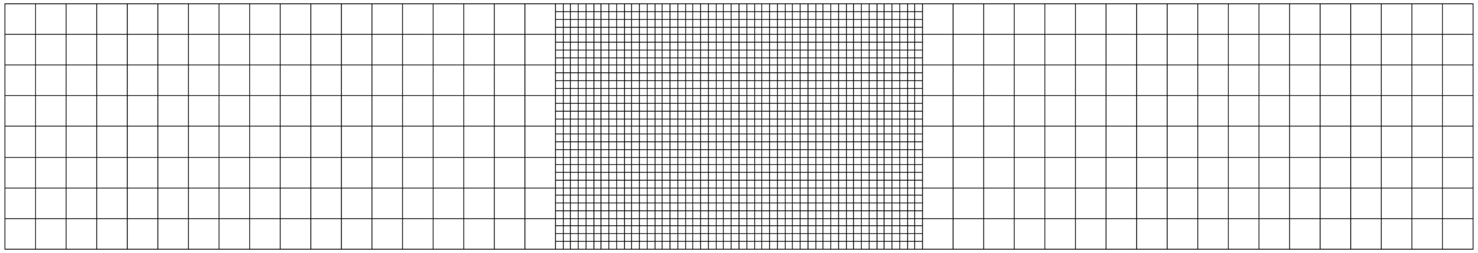
- On the following lock-exchange slides, the lower figure is a zoom in on the center region of the tank.

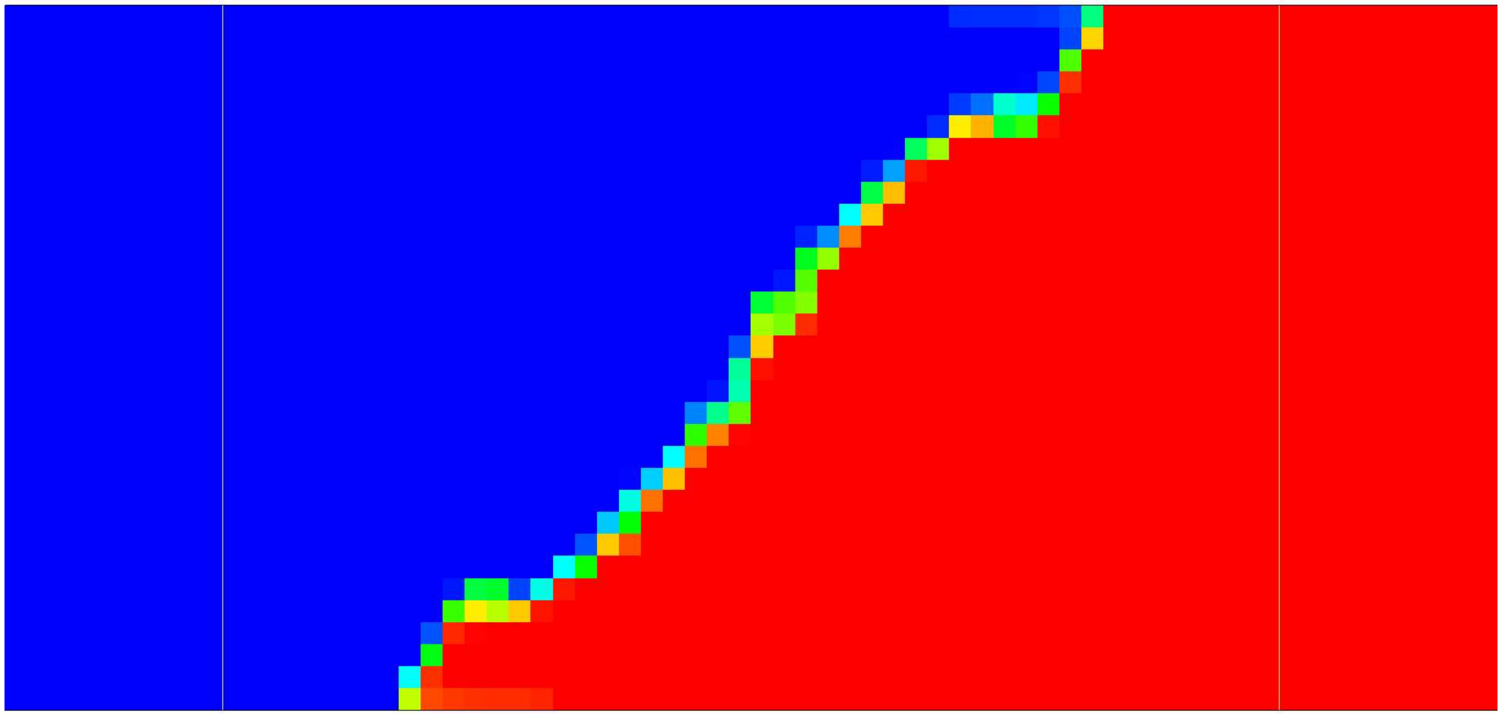
Lock-Exchange: Why is AMR important?



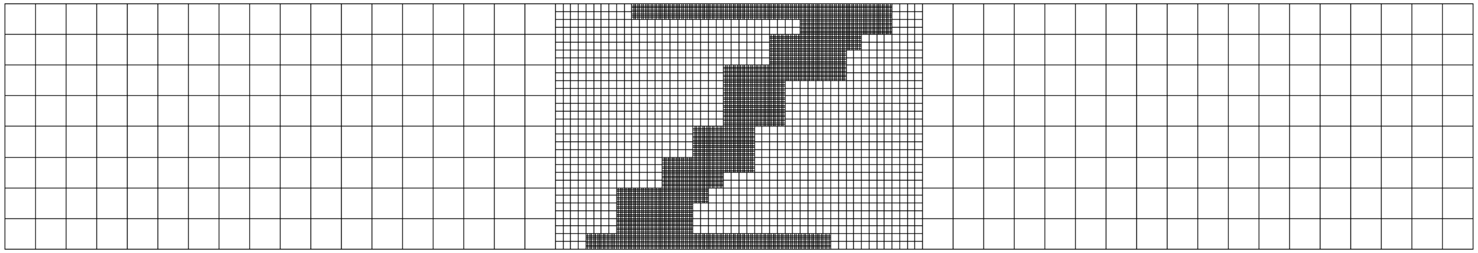


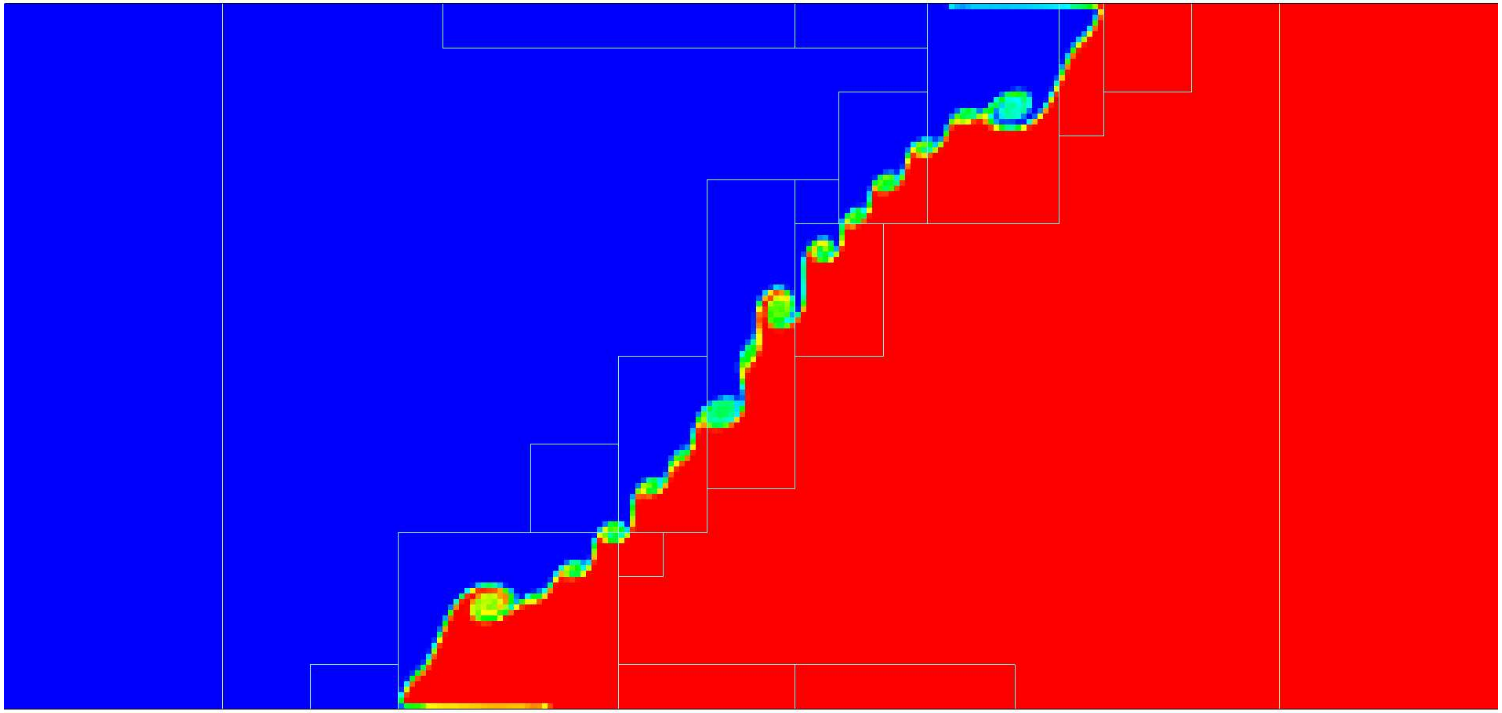
Lock-Exchange: Why is AMR important?





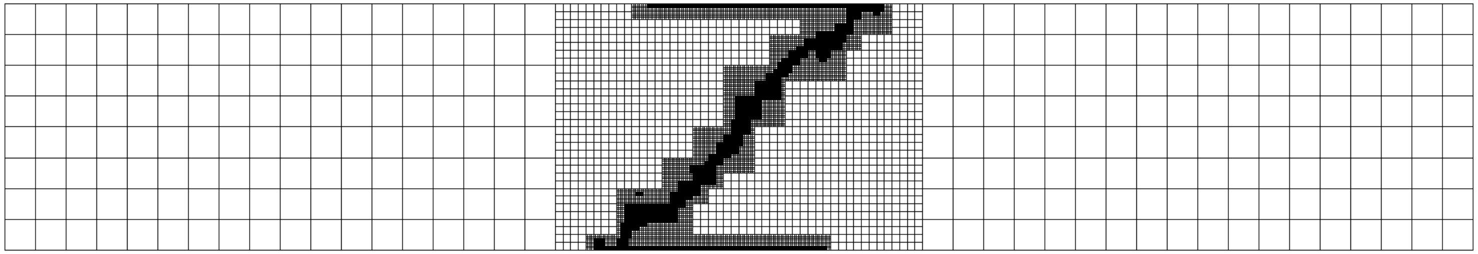
Lock-Exchange: Why is AMR important?

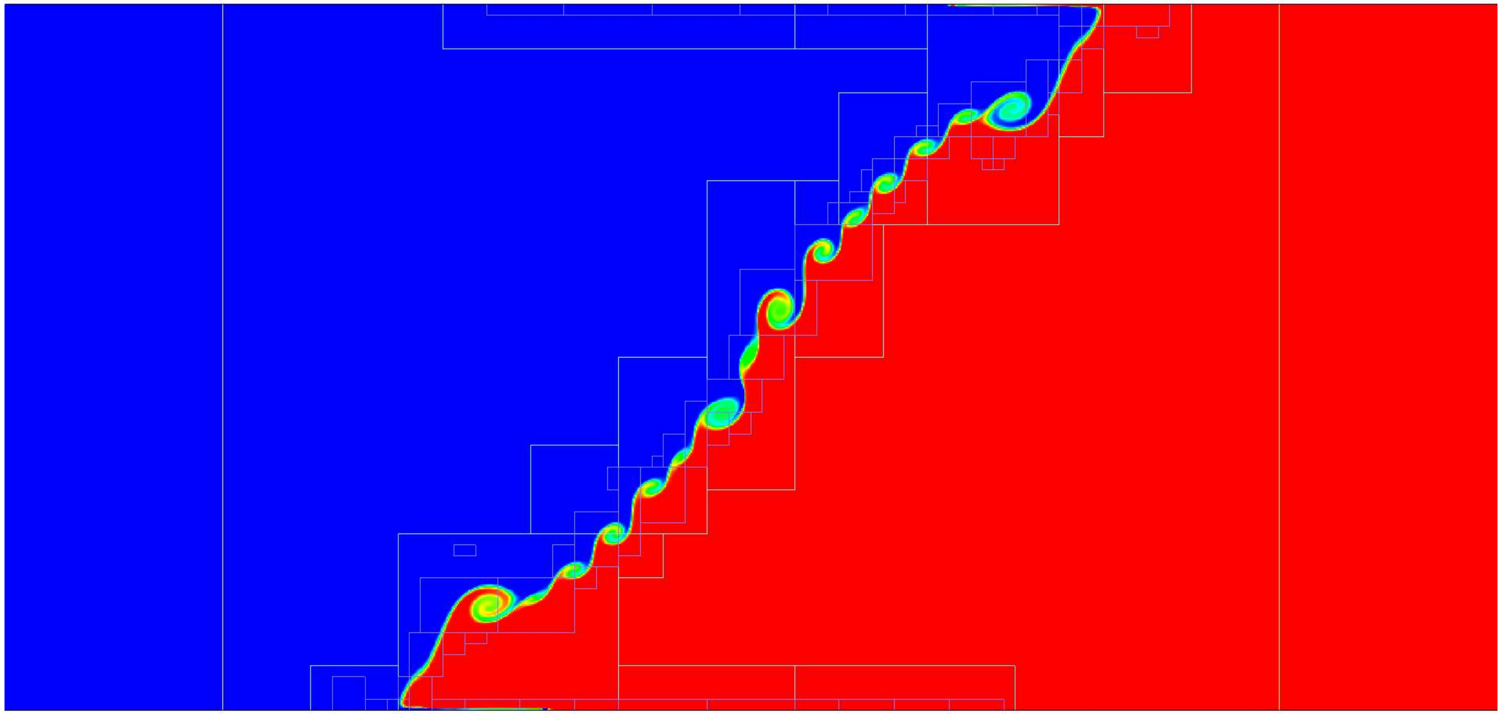




Lock-Exchange: Why is AMR important?

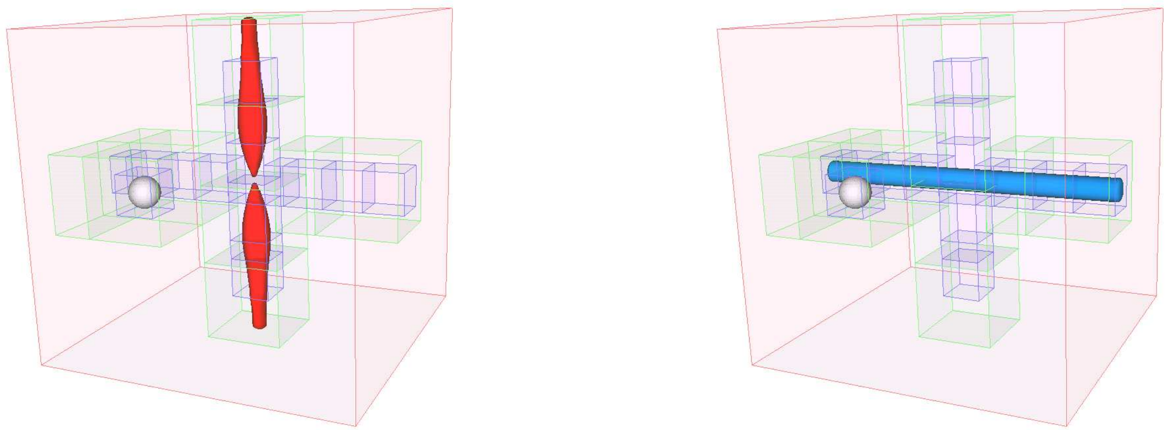
Answer: We can add computational effort only where we need it!





Results: Simulation of flow past a sphere with AMR

- Flow is inside a 1m cubed tank, with a sphere of diameter 0.1m placed at (0.25,0.25,0.5)
- We initialize the flow with a vortex patch in the center of the domain, see below left for z-vorticity iso-surfaces. The $Re = 100$ for this flow. We initialize a passive scalar, see below right for an iso-surface.



Conclusions and Future Work

- We now have a second-order accurate incompressible Navier-Stokes code that has been validated in 2D and 3D.
- Our AMR version is showing reasonable results and is under review to ensure second-order accuracy.
- Future Work:
  - Free Surface Tracking
  - LES Turbulence Closure
  - Field Scale Applications
  - Fourth-Order Accuracy (see my upcoming JCP paper with P. Colella).

Acknowledgments

- U.C. Davis: Professor Schladow, my advisor.
- Lawrence Berkeley National Laboratory: This work is conducted as a close collaboration with Phil Colella and his Applied Numerical Algorithms Group (ANAG). Many thanks to the ANAG staff: Phil, D. Graves, T. Ligocki, D. Martin, P. Schwartz, D. Serafini, G. Smith, T. Sternberg, and B. Van Straalen. This research builds on ANAG's Chombo numerical library.
- This research is funded by the Computational Science Graduate Fellowship program of the Department of Energy.

- Check out my dusty web site: http://edl.engr.ucdavis.edu/barad.html